Wiley
International Journal of Intelligent Systems
Volume 2024, Article ID 6643018, 17 pages
https://doi.org/10.1155/2024/6643018

WILEY

Research Article

Fine-Grained Passenger Load Prediction inside Metro Network via Smart Card Data

Xiancai Tian,[1] **Chen Zhang**,[2] **and Baihua Zheng**[3]

[1]*Living Analytics Research Centre, School of Computing and Information Systems, Singapore Management University, Singapore*
[2]*Department of Industrial Engineering, Tsinghua University, Beijing, China*
[3]*School of Computing and Information Systems, Singapore Management University, Singapore 188065*

Correspondence should be addressed to Chen Zhang; zhangchen01@tsinghua.edu.cn

Received 22 December 2022; Revised 22 May 2024; Accepted 7 June 2024

Academic Editor: Youxi Wu

Metro system serves as the backbone for urban public transportation. Accurate passenger load prediction for the metro system plays a crucial role in metro service quality improvement, such as helping operators schedule train timetables and passengers plan their trips. However, existing works can only predict low-grained passenger flows of origin-destination (O-D) paths or inflows/outflows of each station but cannot predict passenger load distribution over the whole metro network. To this end, this paper proposes an end-to-end inference framework, PIPE, for passenger load prediction of every metro segment between two adjacent stations, by only utilizing smart card data. In particular, PIPE includes two modules. The first is the core. It formulates the travel time distribution of each metro segment as a truncated Gaussian distribution. Since there might be several possible routes for certain O-D paths, the population-level travel time distribution of these O-D paths would be a mixture of travel times of different routes. Considering the route preference may change over time, a dynamic truncated Gaussian mixture model is proposed for parameter inference of each truncated Gaussian distribution of each metro segment. The second module serves as the supplement, which compiles a bunch of methods for predicting passenger flows of O-D paths. Built upon them, PIPE is able to predict the travel time that future passengers of each O-D path will take for passing each metro segment and consequently can predict the passenger load of each metro segment in the short future. Numerical studies from Singapore's metro system demonstrate the efficacy of our method.

1. Introduction

Metro system is the backbone of urban human mobility, especially for metropolises. As the urban population grows, the ever-increasing demands for public transport, especially during peak hours, bring concerns regarding both metro operation security and passenger safety. In such cases, real-time prediction of passenger load over the metro system is highly desirable for both system operators and passengers. It can provide real-time traffic information, help train operation planning, and detect abnormal passenger flow in time.

However, most of the existing works focus on analyzing boarding or alighting passenger load, i.e., inflow or outflow, for each station, or passenger flow of each O-D path at an aggregate low-granularity level. Yet works providing finer-granularity passenger load estimation for each metro segment, i.e., on-board passengers for each rail segment between adjacent stations across the whole network, are limited. The primary reason for this limitation lies in the lack of dependable data sources that effectively capture passengers' movements within metro systems. For instance, while smart card data stands as a prominent form of intelligent transport system (ITS) data utilized for inferring and predicting passenger flows [1], it solely records trip details such as boarding/alighting stations and timestamps, thus failing to monitor passengers' precise real-time whereabouts within the metro infrastructure. Furthermore, for each O-D path, there may be multiple routes. Yet which particular route is taken by a trip is unknown. This brings additional challenges for segmentwise passenger load prediction. Though there

exist few works considering using other types of data, such as cellular data [2], query logs data [3], train weight measurements [4], or camera data on trains [5], for in-train passenger load prediction, these data are not generally available in most metro systems and require additional costs for sensor allocation and data collection.

To address the challenges mentioned above, this paper proposes an end-to-end inference framework, namely *PIPE*, for fine-grained passenger load prediction for each metro segment. *PIPE* only requires smart card data for analysis, and as a tradeoff, does not target at prediction of exact onboard passenger counts for each train. Instead, it aims at predicting real-time "virtual" passenger load for each metro segment. These passengers can be actually located on different trains, if there is more than one train in one segment during one prediction time window. Particularly, *PIPE* has two modules. The first is the core. It formulates the travel time distribution of each metro segment as a truncated Gaussian distribution. Since there might be several possible routes for certain O-D paths, the population-level travel time distribution of these O-D paths would be a mixture of travel times of different routes. Consider the route preference may further change over time. A dynamic truncated Gaussian mixture model is proposed for parameter inference of each truncated Gaussian distribution of each metro segment. The second module serves as the supplement. It compiles a bunch of methods for online passenger flow prediction of each O-D path. Built upon them, *PIPE* is able to predict the travel time that future passengers of each O-D path will take for passing each metro segment, and hence can predict the passenger load of each metro segment in the short future.

The remainder of this paper is organized as follows. Section 2 reviews state-of-the-art works related to this paper. Section 3 presents the model setup, including trip reconstruction and route choice set generation. Section 4 introduces our proposed *PIPE* in detail. Section 5 reports case study results using smart card data collected from the Singapore metro system. Finally, Section 6 provides some concluding remarks.

2. Literature Review

Research topics related to our work include (i) passenger flow prediction for inflow/outflow of each station or O-D matrix of all the paths and (ii) travel time estimation and route selection. State-of-the-art methods in these topics are carefully reviewed as follows.

2.1. Passenger Flow Prediction. Passenger flow prediction has been extensively studied in many literature works, utilizing a variety of ITS datasets such as smart card data and cellular data. For example, some pioneer works adopt time series models, such as the autoregressive integrated moving average (ARIMA) model [6]. Later, some online decomposition-based methods, such as the nonnegative matrix factorization model [7] and wavelet decomposition model [8], have also been proposed, where the passenger flow features are linearly extracted and used for prediction. To extract more complex spatial and temporal features for traffic flow prediction, deep neural network-based methods have been emergingly developed. Various network structures have been proposed, such as stacked autoencoder [9], long short-term memory (LSTM) [10], sequence-to-sequence model with attention mechanism [11], and graph convolutional network (GCN) [12]. Recent works also consider multistep station-level flow prediction based on spatiotemporal hierarchical attention mechanism [13] and the graph transformer model [14]. However, all these models in this topic only target at inflow/outflow prediction for each station separately, regardless of where the passengers come from or where they are headed, let alone how they are distributed inside the metro system.

The O-D matrix prediction aims at predicting trip demands between each O-D path, where deep learning-based methods have become the mainstream. Some typical examples, such as [15], apply LSTM to predict the O-D matrix in the metro system. To better address the spatial dependence of passenger flows, the authors in [16] formulated the O-D matrix together with other geographical features as tensors and developed a multiscale convolutional LSTM for prediction. By considering traffic networks which were naturally graph-structured, the authors in [17] proposed a multiperspective GCN with LSTM to extract temporal features for the O-D matrix. The authors in reference [18] also proposed a matrix factorization-embedded GCN for prediction. The authors in references [19, 20] further considered that the dependence relationships of stations were dynamic over time and proposed dynamic GCN. Yet these methods treat the transport system as a black box and only predict the number of passengers entering or exiting the system for each O-D path but cannot track the passengers' route choices or real-time locations inside the system. Consequently, they cannot estimate the passenger load for each metro segment accurately.

In recent years, several studies have endeavored to make fine-grained passenger load predictions by leveraging alternative data sources. For instance, the authors in reference [2] focused on short-term trajectory prediction (StTP) for individual metro passengers, achieved through the identification of diverse mobility patterns from Wi-Fi probe data. In reference [3], a system was proposed to forecast the train occupancy levels in the near future using crowd-sourced data, where passengers specify their departure and arrival stations, train departure times, and categorize train occupancy as low, medium, or high. The authors in reference [4] utilized load data derived from weight measurements in the air suspension system of train cars to estimate in-train passenger numbers, assuming an average passenger weight of 78 kg. Furthermore, the authors in reference [5] introduced a deep learning methodology for person localization, tracking, and counting within public transport settings, drawing from real-scale CCTV recordings in a laboratory environment. Nevertheless, these data sources are not universally accessible across most metro systems and necessitate additional expenses for sensor deployment and data acquisition, thereby constraining their practical utility in real-time inference for metro-segment flow prediction.

2.2. Travel Time Estimation and Route Selection. To better track passengers' trajectories inside the metro system, many recent research studies rely on probabilistic generative models to infer travel time distribution and passengers' route choice behaviors. These models generally adopt a decomposition framework, i.e., decomposing the travel time of each candidate route into time of its included transit links: the platform walking time, train waiting time, in-vehicle travel time, transfer time, etc. Then, the travel time of a route is the sum of the time of all its transit links, and its total travel time distribution can be inferred from the travel time distribution of each transit link. In particular, the authors in [21] assumed that the time of each transit link is fixed and could be simply derived based on every historical trip's travel time of each O-D path. The authors in reference [22] classified trips into one-transfer, two-transfer, and non-transfer trips and estimated the travel time in different cases. These methods cannot deal with cases with multiple route choices. To take this into account, the authors in [23] further assumed the probabilities of choosing each route as a multinomial distribution. However, the abovementioned methods only take several O-D paths as examples and are hard to scale to the whole metro network-wide estimation. Recently, the authors in [24] formulated the transit links of all the O-D paths as a graph and proposed a computational graph approach for link travel time estimation. However, all the abovementioned methods treat the link travel time as a determined value which may lead to information loss and also hinder their application for probabilistic passenger load prediction of each metro segment.

There also exist some works considering the randomness of travel time by assuming that travel time follows a certain distribution [25–27]. This kind of method is generally cooperated together with route choice selection models. For example, the authors in [27] assumed Gaussian distribution on the transit links and estimated the distributions' parameters by historical trip data whose O-D paths have only a single route. Based on the estimation, route preferences of trip data whose O-D paths have multiple routes can be easily conducted. However, this two-step model has information loss. The authors in reference [25] proposed a unified model by assuming the route preference as a logistic regression of attribute data. The regression parameters and Gaussian parameters are jointly estimated via a Markov chain Monte Carlo method. The authors in reference [26] considered that the logistic regression parameters were further related to train congestion status, which was calculated via additional on-train sensors for measuring passenger volume. In summary, all the existing methods either only consider limited O-D paths or require additional supplementary operation data for inference.

3. Model Setup

We first propose a trip reconstruction process in Section 3.1, which decomposes a trip into a sequence of transit links. Then, in Section 3.2, we formulate the metro system as an undirected network and generate a feasible route set for each O-D path. These two steps lay foundation for *PIPE*. For presentation convenience, Table 1 lists the notations that are frequently used in the rest of the paper.

3.1. Trip Reconstruction. We model a metro network as a graph $G(\mathcal{S}, \mathcal{E}, \mathcal{L})$, consisting of a set of metro stations $\mathcal{S}$, a set of edges $\mathcal{E}$, and a set of metro lines $\mathcal{L}$. A station $s \in \mathcal{S}$ could be either a *normal station* that is crossed by only one metro line or an *interchange station* that is crossed by multiple lines. An edge (or a segment, interchangeably) $e(s_i, s_j, l_k) \in \mathcal{E}$ is defined as a segment on a metro line $l_k \in \mathcal{L}$ that connects two adjacent stations s_i and s_j without passing any other station. Stations s_i and s_j are adjacent if there is an edge $e(s_i, s_j, l_k) \in \mathcal{E}$ between them. Note that there could be multiple edges between two adjacent stations s_i and s_j, corresponding to different metro lines. Our formulated network is undirected, which is reasonable, since most metro systems in the world are bidirectional. However, our proposed method could also be easily extended to cases where a metro system has single-directional lines and should be modelled as a directed graph.

A route r_{ij}^m from an origin station s_i to a destination station s_j is a sequence of L_{ij}^m adjacent edges $\mathcal{L}_{ij}^m = \left\{e_1, \ldots, e_{L_{ij}^m}\right\}$ that could bring passengers from s_i to s_j. In this paper, we only consider simple routes without loop, so each route only visits a station at most once. We denote T_{ij}^m as the corresponding travel time required when a passenger takes a particular route r_{ij}^m to travel from s_i to s_j (note that there could be multiple possible routes which will be detailed later). As illustrated in Figure 1, a trip normally consists of several links: the *entry link*, several *travel links*, and the *exit link*. If the route requires transfers, additional one or more *transfer links* are involved. Accordingly, we can model T_{ij}^m by decomposing it into travel time of the different links described above.

(1) T_i^g represents the time required by an entry link, consisting of the walking time from an entry turnstile at the origin station s_i to the platform and the waiting time for the next train at the platform.

(2) T_e^o represents the time required by a travel link, i.e., the time spent travelling on edge e.

(3) T_s^q represents the time required by a transfer link, consisting of the walking time from one metro platform to another, and the waiting time for the next train at an interchange station s.

(4) T_j^a represents the time required by an exit link, i.e., the walking time from the platform to the turnstiles at the destination station s_j.

Hereafter, the term *transit link* is used to refer to one link of any type required by a trip. The sum of the travel time of all the transit links finally forms the total travel time required by a trip from s_i to s_j by route r_{ij}^m, i.e.,

$$T_{ij}^m = T_i^g + \sum_{e \in \mathcal{L}_{ij}^m} T_e^o + \sum_{s \in \mathcal{S}_{ij}^m} T_s^q + T_j^a, \tag{1}$$

Table 1: Notations of the frequently used variables in *PIPE*.

Variable	Definition
$G(\mathcal{S}, \mathcal{E}, \mathcal{L})$	A general transportation graph with $\mathcal{S}$, $\mathcal{E}$,*and* $\mathcal{L}$ representing the sets of metro stations, edges, and lines, respectively
$e(s_i, s_j, l_k)$	An edge in $\mathcal{E}$ on a metro line $l_k \in \mathcal{L}$ that connects two adjacent stations s_i and s_j
$\mathcal{R}_{ij}$	The set of candidate routes corresponding to a given O-D path $\langle s_i, s_j \rangle$
M_{ij}	The cardinality of $\mathcal{R}_{ij}$
r_{ij}^m	The m^{th} candidate route out of $\mathcal{R}_{ij}$
T_{ij}^m	Travel time required by route r_{ij}^m
T_i^g	Travel time required by the entry link at station s_i
T_e^o	Train travel time corresponding to an edge e
T_u^q	Transfer time required at the interchange station s_u
T_j^a	Travel time required by the exit link at station s_j
$\mathcal{S}_{ij}^m$	The interchange station set of the route r_{ij}^m
$\mathcal{L}_{ij}^m$	The edge set of the route r_{ij}^m
L_{ij}^m	The number of included edges on the route r_{ij}^m, i.e., the cardinality of L_{ij}^m
$\mathcal{K}_{ij}^m$	The transit link set of the route r_{ij}^m
$\pi_{ij}^{m,c}(\tau)$	The probability that passengers of category c entering station s_i at time τ choose route r_{ij}^m out of $\mathcal{R}_{ij}$
tr	A trip record is captured by the smart card data, in the form of $tr(id, o, d, \tau, T, c)$, where id is an encrypted unique string identifying the smart card ID of the trip, where o refers to its origin station, d refers to its destination station, τ records the timestamp when the passenger enters o, T refers to its total travel time, i.e., the difference between timestamps of leaving d and entering o, and $c \in \mathcal{C}$ refers to the card type identifying passenger category (e.g., $\mathcal{C}$ = {child, adult, elderly, student} in Singapore)

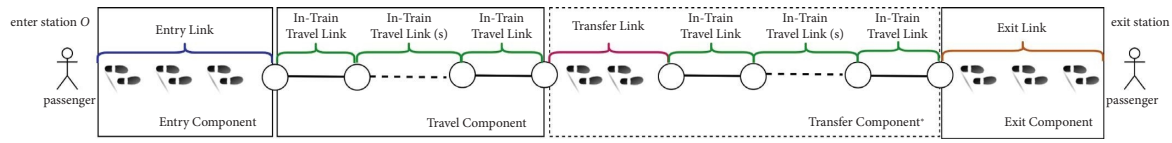


Figure 1: Transit links of trips in a metro system.

where $\mathcal{S}_{ij}^m$ refers to the set of interchange stations on route r_{ij}^m where passengers make transfers. If we can calculate the travel time required by each transit link involved in route r_{ij}^m, we can predict the alighting time of a trip, given its boarding time. In addition, we can also infer the position of a passenger in the metro system at any time point before he/she ends the trip.

It is to be noted that for edge $e(s_i, s_j, l_k)$, we assume that the travel time required from s_i to s_j via metro line l_k is identically distributed as that required from s_j to s_i via the same line. This assumption generally holds for most metro systems. Yet our analytic framework could also be easily extended to cases when the travel time from s_i to s_j is asymmetric, even along the same metro line.

3.2. Route Choice Set Generation. Commonly, in a metro system, there could be multiple routes for some O-D paths. In the following, the term *route choice set* corresponding to each O-D path from s_i to s_j, denoted as $\mathcal{R}_{ij}$, represents all the routes used by passengers travelling from s_i to s_j. $\mathcal{R}_{ij}$ can be achieved either from metro operators or surveys from passengers. It can also be generated in a data-driven way, such as brute-force search and edge elimination, using the topological structure of the metro network. In this paper, considering that the number of stations in a metro system is usually not too large, the simple brute-force search algorithm can be adopted with affordable computation time to generate $\mathcal{R}_{ij}$ for different O-D paths. In particular, note that not all the available routes are actually feasible, since passengers do not prefer a route that is much longer or requires much more transfers than others. We exclude routes that have any of the following characteristics: (i) routes with any loops; (ii) routes that are not the shortest in terms of number of transit links but require more than σ transfers; and (iii) routes whose number of transit links is β ($> rbin1$) times more than the number of transit links of the shortest route. The controlling parameters β and σ could be set according to the assumptions of passengers' behavior. In our study, we set both β and σ as 2, which are big enough to make sure that all the reasonable routes from s_i to s_j are included in $\mathcal{R}_{ij}$.

4. Fine-Grained Passenger Load Prediction

In this section, we propose a framework, namely, *PIPE* for fine-grained prediction of passenger load inside the metro network based on smart card data. In particular, given a metro system $G(\mathcal{S}, \mathcal{E}, \mathcal{L})$, for a particular day, we would like to predict the passenger load on edge e at some time point t in the future, i.e., $X_e(t)$, based on the smart card data. This can be formulated as

$$X_e(t) = \sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{S}\wedge j\neq i} \int_{\tau<t} V_{ij}(\tau) P_{e|ij}(\tau) P\left(T_{i\longrightarrow e} = t-\tau\right) d\tau, \tag{2}$$

where $V_{ij}(\tau)$ represents the number of passengers boarding at s_i at time point τ and later alighting at s_j; $P_{e|ij}(\tau)$ represents the probability that a passenger boarding at s_i and alighting at s_j would take a route containing edge e, given its boarding time τ; and $P(T_{i\longrightarrow e} = t-\tau)$ represents the probability that it takes the time duration $t-\tau$ for a passenger boarding at s_i to reach edge e.

Similarly, we can model the number of passengers alighting at s_j in the future time t, i.e., $X_j^{\text{out}}(t)$, as

$$X_j^{\text{out}}(t) = \sum_{i\in\mathcal{S}\wedge i\neq j} \int_{\tau<t} V_{ij}(\tau) P\left(T_{ij} = t-\tau\right) d\tau, \tag{3}$$

where $P(T_{ij} = t-\tau)$ represents the probability that it takes time duration $t-\tau$ for a passenger to travel from s_i to s_j.

To predict $X_e(t)$ and $X_j^{\text{out}}(t)$, we need to infer the distributions of $T_{i\longrightarrow e}$ and T_{ij}, which are actually dependent on the travel time of all the transit links T_i^g, T_e^o, T_s^q, and T_j^a. We also need to infer $P_{e|ij}(\tau)$, which is the probability that an individual passenger will pass by edge e. It depends on the route(s) he actually takes. In this sense, $P_{e|ij}(\tau)$ actually reflects the route choice probabilities of passengers. Furthermore, we need to predict the O-D matrix $V_{ij}(\tau), i, j \in \mathcal{S}$.

We assume to have historical smart card dataset $\mathcal{TR}$ which includes in total N metro trips, i.e., $\mathcal{TR} = \{tr_n, n = 1, \ldots, N\}$, to infer all these required variables mentioned above. Each metro trip is represented as $tr_n(id_n, o_n, d_n, \tau_n, T_n, c_n)$, where id_n is an encrypted unique string identifying the smart card ID of trip n; o_n refers to the origin station of trip n; d_n refers to the destination station of trip n; τ_n records the timestamp when the passenger enters o_n; T_n refers to the total travel time of trip n, i.e., the difference between timestamps of leaving d_n and entering o_n; and $c_n \in \mathcal{C}$ refers to the card type identifying passenger category (e.g., $\mathcal{C}$ = {child, adult, elderly, student} in Singapore). Note that, in reality, data are recorded in discrete time points $\{t = 1, 2, \ldots\}$. We have to break continuous time into discrete windows and replace the integration in (2) and (3) and by the corresponding discrete summation. Without loss of generality, in this paper, we break continuous time into 20-minute time windows.

In the following, we will introduce *PIPE*, which includes two modules: (i) predicting the number of passengers travelling between each O-D path given the entry time τ, i.e., $V_{ij}(\tau)$ and (ii) estimating the travel time distributions for all transit links, i.e., T_i^g, T_e^o, T_s^q, and T_j^a and route choice probabilities for all the candidate routes, i.e., $P_{e|ij}$. Based on (i) and (ii), we can conduct online predictions for (2) and (3). Figure 2 plots the architecture of *PIPE*.

4.1. O-D Matrix Prediction. As reviewed in Section 2, predicting $V_{ij}(\tau)$ (here without confusion, we omit the day subscription for brevity) has been thoroughly studied in the literature. In our paper, we consider the following methods in our pool and suggest selecting the one with the best prediction results. In particular, the calendar model, ARIMA model, and LSTM model are time series models that analyze temporal traffic patterns of station pairs using historical data to forecast future traffic trends. Conversely, linear regression models and random forest models leverage both temporal and spatial information by incorporating historical traffic volumes from specific station pairs and the inbound and outbound traffic of neighboring stations. Of course, other more advanced methods as reviewed in Section 2 can also be compiled in the pool depending on the practitioner's preference.

(1) *Calendar Model.* We simply use the historical average of $V_{k,ij}(\tau)$ of the last $k = 1, \ldots, K$ days to predict $V_{ij}(\tau)$ for the next new day, i.e.,

$$V_{ij}(\tau) = \frac{\sum_{k=1}^{K} V_{k,ij}(\tau)}{K}. \tag{4}$$

(2) *Linear Regression Models.* For each O-D path, we predict $V_{ij}(\tau)$ based on past passenger inflow $X_s^{\text{in}}(\tau - l)$ and passenger outflow $X_s^{\text{out}}(\tau - l)$ at time $t - l$ of station s with a linear regression model, i.e.,

$$V_{ij}(\tau) = \sum_{s\in\mathcal{S}} \sum_{l=1}^{\Delta} \left[a_{ij}^{s,l} X_s^{in}(\tau - l) + b_{ij}^{s,l} X_s^{\text{out}}(\tau - l) \right] + \epsilon_{ij}(\tau), \tag{5}$$

where $a_{ij}^{s,l}$ and $b_{ij}^{s,l}$ are regression coefficients of passenger inflow and outflow at station s with lag order l, respectively; Δ is the maximum lag order decided based on validation performance; and $\epsilon_{ij}(\tau)$ is the noise. Considering that the number of inputs is high, we introduce regularization to the coefficients using l_1 or l_2 penalties, i.e., lasso [28] or ridge regression [29].

(3) *Random Forest Model* [30]. Each decision tree takes lagged passenger inflow $X_s^{in}(t - l)$ and outflow $X_s^{\text{out}}(t - l), l = 1, \ldots, \Delta$ as predictors, and the mean of predictions of all the individual trees forms the final prediction.

(4) *ARIMA Model* [31]. For each $V_{ij}(\tau)$, we formulate an ARIMA (P, d, Q) model as

$$\left(1 - \sum_{p=1}^{P} \phi_{ij,p} L^q\right)(1 - B)^d V_{ij}(\tau) = \left(1 + \sum_{q=1}^{Q} \theta_{ij,q} B^q\right) \epsilon_{ij}(\tau), \tag{6}$$

where B is the lag operator, $\phi_{ij,p}$ are the parameters of the autoregressive part, $\theta_{ij,q}$ are the parameters of the moving average part, and d is the integrated order. For different $V_{ij}(\tau)$, P, d, and Q are determined by the Akaike information criterion.

(5) *LSTM* [32]. The network consists of a LSTM module with 500 hidden units and a fully connected layer. For each $V_{ij}(\tau)$, we use its lagged counterparts $V_{ij}(\tau - l), l = 1, 2, \ldots, \Delta$ as prediction features.

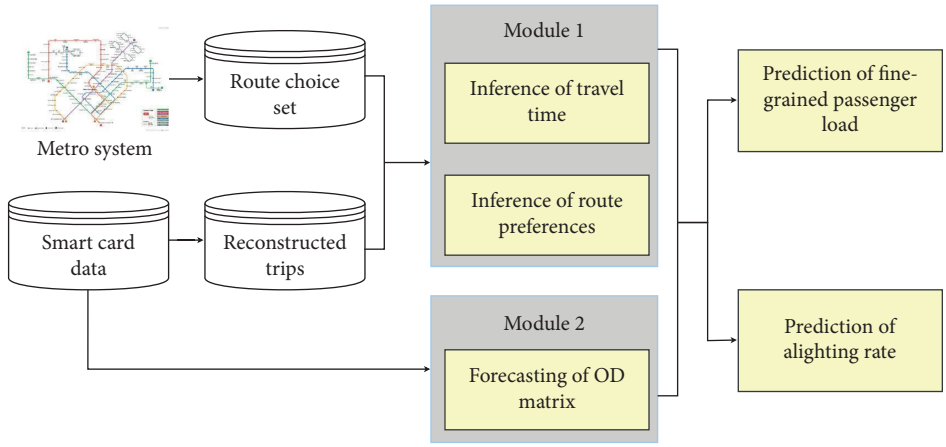


Figure 2: PIPE framework for fine-grained prediction of passenger load inside the metro network.

For ARIMA and LSTM, it is to be noted that when predicting $V_{ij}(\tau)$ for an O-D path with a very long travel time, certain nearest inputs $V_{ij}(\tau-1),\ldots,V_{ij}(\tau-d)$ may be unavailable. This is because passengers boarding at the time window $\tau-1$ to $\tau-d$ have not yet completed their trips, i.e., have not yet tapped out on s_j's turnstile. Consequently, we do not know which station s_j would be. For this case, we suggest removing these d nearest windows' data from the input set. We take the Singapore MRT network as an example. Trips can be finished within two hours. As we consider a 20-minute time window, we have $d=6$. Consequently, we use $V_{ij}(\tau-7), V_{ij}(\tau-8), \cdot s$ as inputs to estimate $V_{ij}(\tau)$ for ARIMA and LSTM.

4.2. Inference of Travel Time and Route Preferences. This is the core of *PIPE*. We discuss how to infer the travel time distributions of transit links and probabilities of route choices. Ideally, the travel time of each transit link would be fixed if the train and passengers have constant speeds. However, in reality, the travel time would have certain variance due to train speed variance. As such, it is reasonable to model the travel time by a Gaussian distribution. However, we consider that speeds can only vary in a certain interval with upper and low limits. We model T_s^g, T_e^c, T_s^q, and T_s^a via truncated Gaussian distributions: $T_s^g \sim TN(\mu_s^g, \sigma_s^g, a_s^g, b_s^g)$, $T_e^o \sim TN\ (\mu_e^o, \sigma_e^o, a_e^o, b_e^o)$, $T_s^q \sim TN\ (\mu_s^q, \sigma_s^q, a_s^q, b_s^q)$, and $T_s^a \sim TN(\mu_s^a, \sigma_s^a,\ a_s^a, b_s^a)$. The probability distribution function of the truncated Gaussian distribution $TN(\mu,\sigma,a,b)$ is defined as

$$x \sim TN(\mu,\sigma,a,b) = \begin{cases} \dfrac{\phi(x-\mu/\sigma)}{\sigma(\Phi(b-\mu/\sigma)-\Phi(a-\mu/\sigma))}, & \text{if } a \le x \le b, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

where $\phi(\cdot)$ is the probability density function of the standard normal distribution and $\Phi(\cdot)$ is its cumulative distribution function.

For a given O-D path $\langle s_i, s_j\rangle$, it has $m=1,\ldots,M_{ij}$ possible routes and each route r_{ij}^m includes L_{ij}^m edges with set $\mathscr{L}_{ij}^m$. The total travel time of a trip taking route r_{ij}^m can be decomposed into the travel time of different transit links, as defined in (1). For notation convenience, we denote all of its transit links as a set $\mathscr{H}_{ij}^m$, and the total travel time equals the sum of the time required by each transit link in $\mathscr{H}_{ij}^m$, i.e.,

$$T_{ij}^m = T_i^g + \sum_{e\in\mathscr{L}_{ij}^m} T_e^o + \sum_{s\in\mathscr{S}_{ij}^m} T_s^q + T_j^a = \sum_{h\in\mathscr{H}_{ij}^m} T_h. \tag{8}$$

Then, the distribution of T_{ij}^m can also be approximated by a truncated Gaussian distribution [33] as

$$T_{ij}^m \sim TN\left(\mu_{ij}^m, \sigma_{ij}^m, a_{ij}^m, b_{ij}^m\right), \tag{9}$$

where

$$\begin{aligned} \mu_{ij}^m &= \mu_i^g + \sum_{e\in\mathscr{L}_{ij}^m} \mu_e^o + \sum_{s\in\mathscr{S}_{ij}^m} \mu_s^q + \mu_j^a = \sum_{h\in\mathscr{H}_{ij}^m} \mu_h, \\ \sigma_{ij}^{m^2} &= \sigma_i^{g^2} + \sum_{e\in\mathscr{L}_{ij}^m} \sigma_e^{o^2} + \sum_{s\in\mathscr{S}_{ij}^m} \sigma_s^{q^2} + \sigma_j^{a^2} = \sum_{h\in\mathscr{H}_{ij}^m} \sigma_h^2, \\ a_{ij}^m &= a_i^g + \sum_{e\in\mathscr{L}_{ij}^m} a_e^o + \sum_{s\in\mathscr{S}_{ij}^m} a_s^q + a_j^a = \sum_{h\in\mathscr{H}_{ij}^m} a_h, \\ b_{ij}^m &= b_i^g + \sum_{e\in\mathscr{L}_{ij}^m} b_e^o + \sum_{s\in\mathscr{S}_{ij}^m} b_s^q + b_j^a = \sum_{h\in\mathscr{H}_{ij}^m} b_h. \end{aligned} \tag{10}$$

For cases considering all the M_{ij} routes, we assume the probability that passengers of card type c entering s_i at time τ would choose the route m out of $\mathscr{R}_{ij}$, which is $\pi_{ij}^{m,c}(\tau)$. Consequently, the travel time distribution of $T_{ij}(\tau)$ will follow a dynamic truncated Gaussian mixture model as follows:

$$T_{ij}^{c}(\tau) \sim \sum_{m=1}^{M_{ij}} \pi_{ij}^{m,c}(\tau) TN\left(\mu_{ij}^{m}, \sigma_{ij}^{m}, a_{ij}^{m}, b_{ij}^{m}\right). \tag{11}$$

There are mainly two reasons for this setting of $\pi_{ij}^{m,c}(\tau)$. First, considering that travel behaviors may vary across passenger types, in this study, we break down the probabilities based on passenger types and suppose $\sum_{m=1}^{M_{ij}} \pi_{ij}^{m,c}(\tau) = 1, \forall c \in \mathscr{C}$. For example, in Singapore, there are four categories of passengers, i.e., $\mathscr{C}$ = {adult, child, elderly, student}, and they have different route preferences. For example, the elderly prefer comfortable routes which may require longer travel time but are less crowded, since old people are usually more flexible in terms of time and yet are physically more vulnerable. In contrast, working adults, who generally rush for time, probably prefer short routes with small time costs, even though they could be overcrowded. Second, the probabilities may change over time. For example, during commuting hours, the elderly may prefer longer yet uncrowded routes. However, after commuting hours, the short routes will also become uncrowded. Then, their preference may change to these short and uncrowded routes.

We denote the parameter set of the abovementioned dynamic truncated Gaussian mixture model as $\boldsymbol{\Theta}$, which includes $\{\mu_s^g, \sigma_s^g\}$, $\{\mu_s^q, \sigma_s^q\}$, and $\{\mu_s^a, \sigma_s^a\}$ for any station $s \in \mathcal{S}$, $\{\mu_e^c, \sigma_e^c\}$ for any edge $e \in \mathscr{E}$, and $\left\{\pi_{ij}^{m,c}(\tau)\right\}$ for any O-D path $\langle s_i, s_j \rangle$ and passenger category $c \in \mathscr{C}$. Now, we use the accumulated smart card data $\mathscr{TR} = \{tr_n, n = 1, \ldots, N\}$ to estimate $\boldsymbol{\Theta}$. Here, we adopt a Bayesian estimation framework. In particular, to capture the dynamics of $\pi_{ij}^{m,c}(\tau)$ over time, we assume it has a Dirichlet distribution with parameter $\pi_{ij}^{m,c}(\tau - 1)$ as prior, i.e.,

$$\begin{aligned} p_0\left(\pi_{ij}^{m,c}(\tau), m = 1, \ldots, M_{ij}\right) &= \text{Dirichlet}\left(\pi_{ij}^{m,c}(\tau - 1), m = 1, \ldots, M_{ij}\right) \\ &= \frac{1}{B\left(\left[\pi_{ij}^{1,c}(\tau - 1), \ldots, \pi_{ij}^{M_{ij},c}(\tau - 1)\right]\right)} \prod_{m=1}^{M_{ij}} \pi_{ij}^{m,c}(\tau)^{\pi_{ij}^{m,c}(\tau-1)-1}, \end{aligned} \tag{12}$$

where $B([\pi_{ij}^{1,c}(\tau - 1), \ldots, \pi_{ij}^{M_{ij},c}(\tau - 1)]) = \prod_{k=1}^{M_{ij}} \Gamma(\pi_{ij}^{1,c}(\tau - 1))/\Gamma(\sum_{k=1}^{M_{ij}} \pi_{ij}^{1,c}(\tau - 1))$ is the beta function. For all the other parameters, we assume uniform prior distributions. Hence, $p_0(\boldsymbol{\Theta}) = \prod_{i,j \in \mathcal{S}, M_{ij} > 1} \prod_{c \in \mathscr{C}} \prod_{\tau=1}^{T} p_0(\pi_{ij}^{m,c}(\tau), m = 1, \ldots, M_{ij})$. Then, we use the maximum a posteriori (MAP) method in the Bayesian estimation framework to estimate $\boldsymbol{\Theta}$. MAP aims to find the optimal $\boldsymbol{\Theta}$ such that the posterior $p(\boldsymbol{\Theta} \mid \mathscr{TR}) \propto p(\mathscr{TR} \mid \boldsymbol{\Theta}) p_0(\boldsymbol{\Theta})$ can be maximized.

However, for trips whose O-D path has more than one possible route, the route choice information is missing; hence the direct maximization of the posterior distribution is difficult. Alternatively, we treat the route choice as an unobserved latent variable and adopt the expectation-maximization (EM) method for estimation. In particular, we denote the missed route choice information for trip tr_n as $\mathbf{Z}_n \in R^{M_n}$ where M_n is the number of routes for the O-D path $\langle o_n, d_n \rangle$. In specific, $\mathbf{Z}_n = [Z_{n1}, \ldots, Z_{nM_n}]$ where Z_{nm} = 1 if the route taken by tr_n is $r_{o_n d_n}^m$ and $Z_{nm} = 0$ otherwise. When only one route is available, $Z_n = 1$ is a scalar. Then, we have $\mathscr{Z} = \{\mathbf{Z}_n, n = 1, \ldots, N\}$. Consequently, the augmented complete posterior $\boldsymbol{\Theta}$ given the AFC card dataset $\mathscr{TR}$ and route choice $\mathscr{Z}$ can be formulated as $p(\boldsymbol{\Theta} \mid \mathscr{TR}, \mathscr{Z})$. Then, the EM algorithm estimates $\boldsymbol{\Theta}$ and $\mathscr{Z}$ alternatively in an iterative way. In the E step of the r^{th} iteration, the expectation of the augmented complete posterior is calculated using the observed data $\mathscr{TR}$ and the current best estimation $\boldsymbol{\Theta}^{r-1}$. In the M step, the parameters are updated as $\boldsymbol{\Theta}^r$ by maximizing the expected augmented completed posterior. These two steps iterate until convergence and obtain $\widehat{\boldsymbol{\Theta}}$ by generating a sequence of parameters $\{\boldsymbol{\Theta}^r\}$ from an initial estimation $\boldsymbol{\Theta}^0$. The specific E and M steps for the r^{th} iteration in our case are shown as follows.

Consider that the augmented completed posterior is

$$\begin{aligned} p(\boldsymbol{\Theta} \mid \mathscr{TR}, \mathscr{Z}) &\propto p(\mathscr{TR}, \mathscr{Z} \mid \boldsymbol{\Theta}) p_0(\boldsymbol{\Theta}) \\ &\propto \prod_{n=1}^{N} \left[I_{M_n > 1} \prod_{m=1}^{M_n} \left(\left(\pi_{o_n,d_n}^{m,c}(\tau_n)\right)^{\pi_{o_n,d_n}^{m,c}(\tau_n - 1)} TN\left(T_n \mid \mu_n^m, \sigma_n^m, a_n^m, b_n^m\right) \right)^{Z_{nm}} + I_{M_n = 1} TN\left(T_n \mid \mu_n, \sigma_n, a_n, b_n\right) \right]. \end{aligned} \tag{13}$$

For label convenience, we abuse the notations $\mu_n^m \equiv \mu_{o_n d_n}^m = \sum_{h \in \mathscr{H}_{o_n d_n}^m} \mu_h$, $\sigma_n^{m2} \equiv \sigma_{o_n d_n}^{m^2} = \sum_{h \in \mathscr{H}_{o_n d_n}^m} \sigma_h^2$, $a_n^m \equiv a_{o_n d_n}^m = \sum_{h \in \mathscr{H}_{o_n d_n}} a_h$, $b_n^m \equiv b_{o_n d_n}^m = \sum_{h \in \mathscr{H}_{o_n d_n}^m} b_h$, and $\pi_n^{m,c_n} \equiv \pi_{o_n d_n}^{m,c_n}$.

In the E step, we evaluate the conditional expectation of the log augmented completed posterior as

$$\begin{aligned} Q\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{r-1}\right) &= E\left[\log\left(p\left(\mathcal{TR}, \mathcal{Z} \mid \boldsymbol{\Theta}\right) p_0\left(\boldsymbol{\Theta}\right)\right) \mid \mathcal{TR}, \boldsymbol{\Theta}^{r-1}\right] \\ &\propto \sum_{n=1}^{N} \left\{ I_{M_n > 1} \sum_{m=1}^{M_n} \widetilde{Z}_{nm} \left[\pi_n^{m,c_n}\left(\tau_n - 1\right) \ln\left(\pi_n^{m,c_n}\left(\tau_n\right)\right) + \ln\left(TN\left(T_n \mid \mu_n^m, \sigma_n^m, a_n^m, b_n^m\right)\right)\right] \right. \\ &\left. + I_{M_n = 1} \ln\left(TN\left(T_n \mid \mu_n, \sigma_n, a_n, b_n\right)\right)\right\}, \end{aligned} \tag{14}$$

where

$$\widetilde{Z}_{nm} = E\left(Z_{nm} \mid \mathcal{TR}, \boldsymbol{\Theta}^{r-1}\right) = \frac{\pi_n^{m,c_n(r-1)}\left(\tau_n\right) TN\left(T_n \mid \mu_n^{m(r-1)}, \sigma_n^{m(r-1)}, a_n^{m(r-1)}, b_n^{m(r-1)}\right)}{\sum_{m=1}^{M_n} \pi_n^{m,c_n(r-1)}\left(\tau_n\right) TN\left(T_n \mid \mu_n^{m(r-1)}, \sigma_n^{m(r-1)}, a_n^{m(r-1)}, b_n^{m(r-1)}\right)}. \tag{15}$$

In the M step, we update $\boldsymbol{\Theta}^r = \text{argmax}_{\boldsymbol{\Theta}} Q\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{r-1}\right)$ by updating each parameter $\{\mu_s^g, \sigma_s^g\}$, $\{\mu_s^a, \sigma_s^a\}$, $\{\mu_s^q, \sigma_s^q\}$ for $s \in \mathcal{S}$, $\{\mu_e^o, \sigma_e^o\}$ for $e \in \mathcal{E}$, and $\left\{\pi_{ij}^{1,c}, \ldots, \pi_{ij}^{M_{ij},c}\right\}$ for $i, j \in \mathcal{S}, M_{ij} > 1, c \in \mathcal{C}$ separately. In particular, the part of (14) that relates to $\{\mu_s^g, \sigma_s^g\}$ is

$$\begin{aligned} \widetilde{l}\left(\mu_s^g, \sigma_s^{g^2}\right) = \sum_{n=1}^{N} \left\{ I_{\left(M_n = 1, o_n = s\right)} \ln\left(TN\left(T_n \mid \mu_n, \sigma_n, a_n, b_n\right)\right) \right. \\ \left. + I_{\left(M_n > 1, o_n = s\right)} \sum_{m=1}^{M_n} \widetilde{Z}_{nm} \left[\ln\left(\pi_n^m\right) \ln\left(TN\left(T_n \mid \mu_n^m, \sigma_n^m, a_n^m, b_n^m\right)\right)\right] \right\}. \end{aligned} \tag{16}$$

The maximization of (16) has no closed-form solution. Consequently, we apply gradient descent to update the value of $\left\{\mu_s^g, \sigma_s^{g^2}\right\}$ in an iterative way. Specifically, the first-order derivatives of (16) with respect to μ_s^g and $\sigma_s^{g^2}$ are

$$\begin{aligned} \frac{\partial \widetilde{l}}{\partial \mu_s^g} &= \sum_{n=1}^{N} I_{\left(M_n = 1, o_n = s\right)} \left[\frac{1}{\sigma_n} \frac{\phi\left(b_n - \mu_n/\sigma_n\right) - \phi\left(a_n - \mu_n/\sigma_n\right)}{\Phi\left(b_n - \mu_n/\sigma_n\right) - \Phi\left(a_n - \mu_n/\sigma_n\right)} + \frac{\left(T_n - \mu_n\right)}{\sigma_n^2}\right] \\ &\quad + \sum_{n=1}^{N} I_{\left(M_n > 1, o_n = s\right)} \left[\sum_{m=1}^{M_n} \widetilde{Z}_{nm} \left(\frac{1}{\sigma_n^m} \frac{\phi\left(b_n^m - \mu_n^m/\sigma_n^m\right) - \phi\left(a_n^m - \mu_n^m/\sigma_n^m\right)}{\Phi\left(b_n^m - \mu_n^m/\sigma_n^m\right) - \Phi\left(a_n^m - \mu_n^m/\sigma_n^m\right)} + \frac{\left(T_n - \mu_n^m\right)}{\sigma_n^{m^2}}\right)\right], \\ \frac{\partial \widetilde{l}}{\partial \sigma_s^{g^2}} &= \sum_{n=1}^{N} I_{\left(M_n > 1, o_n = s\right)} \left[\sum_{m=1}^{M_n} \widetilde{Z}_{nm} \left(\frac{1}{2\sigma_n^{m^3}} \frac{\left(b_n^m - \mu_n^m\right)\phi\left(b_n^m - \mu_n^m/\sigma_n^m\right) - \left(a_n^m - \mu_n^m\right)\phi\left(a_n^m - \mu_n^m/\sigma_n^m\right)}{\Phi\left(b_n^m - \mu_n^m/\sigma_n^m\right) - \Phi\left(a_n^m - \mu_n^m/\sigma_n^m\right)}\right.\right. \\ &\quad \left.\left. + \frac{\left(T_n - \mu_n^m\right)^2}{2\sigma_n^{m^4}} - \frac{1}{2\sigma_n^{m^2}}\right)\right] + \sum_{n=1}^{N} I_{\left(M_n = 1, o_n = s\right)} \left[\frac{\left(T_n - \mu_n\right)^2}{2\sigma_n^4} - \frac{1}{2\sigma_n^2}\right. \\ &\quad \left. + \frac{1}{2\sigma_n^3} \frac{\left(b_n - \mu_n\right)\phi\left(b_n - \mu_n/\sigma_n\right) - \left(a_n - \mu_n\right)\phi\left(a_n - \mu_n/\sigma_n\right)}{\Phi\left(b_n - \mu_n/\sigma_n\right) - \Phi\left(a_n - \mu_n/\sigma_n\right)}\right]. \end{aligned} \tag{17}$$

Similarly, we can estimate $\{\mu_e^o, \sigma_e^{o^2}\}$, $\{\mu_s^q, \sigma_s^{q^2}\}$, and $\{\mu_s^a, \sigma_s^{a^2}\}$ for all the transit links.

For $\pi_{ij}^{m,c}(\tau)$, its maximization has a closed-form solution as

$$\pi_{ij}^{m,c}(\tau) = \frac{\sum_{n=1}^{N} I_{(o_n=i,d_n=j,\tau_n=\tau,c_n=c)} \widetilde{Z}_{nm} \pi_{ij}^{m,c}(\tau-1)}{\sum_{m=1}^{M_{ij}} \sum_{n=1}^{N} I_{(o_n=i,d_n=j,\tau_n=\tau,c_n=c)} \widetilde{Z}_{nm} \pi_{ij}^{m,c}(\tau-1)}. \tag{18}$$

Last but not the least, we empirically estimate the truncation points for each transit link as follows. In specific, the truncation points $[a, b]$ for transfer link T_s^q are set as $[0, 2 * (w_0 + I_l)]$, where w_0 is a predefined transfer walking time from one metro platform to another at an interchange station (if more information about each individual station is available, w_0 can be set differently for different stations) and I_l is the train headway of train service l. As to the truncation points for T_s^g, T_s^a, and T_e^o, since they influence each other in a complex way, we estimate their values iteratively by Algorithm 1.

5. Case Study

In this section, we apply *PIPE* in the Singapore Mass Rapid Transit (MRT) system performance evaluation. We first introduce the dataset and data preprocessing steps. Then, we present the performance of *PIPE* for O-D matrix prediction, travel time inference, outflow prediction of each MRT station, and fine-grained passenger load prediction for each segment over the metro system, respectively.

5.1. EZ-Link Card Data. Up to May 2016, the Singapore MRT network, as shown in Figure 3, consisted of 102 stations of 7 MRT lines (including two line extensions), with in total 114 edges between adjacent stations.

The EZ-Link card is the smart card used in Singapore for the payment of public transport trips. 251,089,965 MRT trip records, which were collected during all the working days from January 1 to May 31 in 2016, are utilized as the data source in our study. Four sample examples are listed in Table 2. As introduced in Section 4, each MRT trip is reformulated as $tr(id, o, d, \tau, T, c)$. As the passenger flow patterns of working days are significantly different from the patterns corresponding to weekends and public holidays, we only focus on data related to working days in this case study. However, our framework can be easily applied to provide weekend/public holiday data prediction as well.

5.2. O-D Matrix Prediction. As presented in Section 4.1, here we consider six candidate methods for O-D matrix prediction. We divide the five months' EZ-link data into three disjoint datasets as follows: 70% of the data are used as the training set, 20% are used as the validation set for model selection, and the remaining 10% are used as the testing set for model evaluation. Table 3 reports the *mean square error (MSE)* of the 100 most busy O-D paths, which cover 13.30% of the whole smart card dataset. Three different prediction horizons of the short future, which refer to the length between the current time window and the time window to be predicted, are considered: one-step with 20 min ahead, four-step with 80 min ahead, and six-step with 120 min ahead separately.

Among the various models considered, linear regression models and the random forest model demonstrate superior performance compared to time series models. This suggests that the inbound traffic and outbound traffic of neighboring stations play a crucial role in forecasting future O-D counts for a specific station pair. Furthermore, the random forest model yields more precise predictions than linear regression models, which is likely attributed to its flexibility in capturing nonlinear relationships between predictors and the target variable. Hence, in our following analysis, we adopt random forest to predict $V_{ij}(\tau)$. However, other more advanced methods can also be compiled in the pool for selection, if better prediction performance can be achieved in other datasets. Since this part is the supplement of *PIPE*, we left the exploration for more advanced methods to practitioners case by case.

In addition, as the prediction horizon increases, all the methods get worse performance, except the calendar model. This is reasonable. As the prediction horizon m increases, more accumulated prediction errors in $V_{ij}(\tau), \ldots, V_{ij}(\tau + m - 1)$ are included as model input when predicting $V_{ij}(\tau + m)$, which consequently deteriorates the prediction accuracy. In contrast, the calendar model simply utilizes the historical average as a prediction; hence it is *not* influenced by the prediction horizon.

To better demonstrate the prediction results, we plot the random forest's one-step-ahead prediction results for two selected O-D paths in Figure 4. It is observed that random forest is able to capture small temporal fluctuations of $V_{ij}(\tau)$ much more accurately than the calendar model. We take the O-D path from Boon Lay to Jurong East in Figure 4(a) as an example, and the random forest achieves a MSE of 22.86, which is significantly lower than the MSE of 334.29 achieved by the calendar model. This notable performance gap is similarly manifested in the O-D path from Pasir Ris to Tampines, as depicted in Figure 4(b).

5.3. Travel Time Prediction. Similar to O-D matrix prediction, here we use 70% training data to infer the travel time distribution of all the transit links based on the proposed dynamic truncated Gaussian mixture model.

For demonstration purposes, we first present the estimated travel time distribution of certain transit links. We take the O-D path from Tanah Merah to Changi Airport as an example. The travel time distribution for each of its transit links is shown in Figure 5. There are in total four transit links that contribute to the travel time from Tanah Merah to Changi Airport, including (a) entry link at Tanah Merah, (b) in-train travel link from Tanah Merah to Expo, (c) in-train travel link from Expo to Changi Airport, and (d) exit link at Changi Airport. These fitted truncated Gaussian distributions of the four links show some interesting phenomena consistent with reality. First, the mean of transit link (a), i.e.,

Input: smart card dataset:
$\mathcal{TR} = \{tr_n(id_n, o_n, d_n, \tau_n, T_n, c_n), n = 1, \ldots, N\}$; edge length l_e (unit: km) for $e \in \mathcal{E}$; travel speed limit v_l (unit: km/min) for $l \in \mathcal{L}$;
Output: $a_s^g, b_s^g, a_s^a, b_s^a$ for $s \in \mathcal{S}$; a_e^o, b_e^o for $e \in \mathcal{E}$;
Initialization: $a_s^g \longleftarrow 0, b_s^g \longleftarrow 0, a_s^a \longleftarrow 0, b_s^a \longleftarrow 0$ for $s \in \mathcal{S}$;
$a_e^o \longleftarrow 0, b_e^o \longleftarrow 0$ for $e \in \mathcal{E}$;
(1) **for** $e(s_i, s_j, l_k) \in \mathcal{E}$ **do**
(2) $\quad Y \longleftarrow \{tr_n \in \mathcal{TR} \mid (tr_n.o = s_i \wedge tr_n.d = s_j) \vee (tr_n.d = s_i \wedge tr_n.o = s_j)\}$
(3) $T_{\max} \longleftarrow \max_{tr_n \in Y}(tr_n.T)$
(4) $b_e^o \longleftarrow T_{\max}, a_e^o \longleftarrow l_e/v_l$
(5) $\quad b_{s_j}^g \longleftarrow \max(b_{s_j}^g, T_{\max} - a_e^o)$
(6) $\quad b_{s_i}^a \longleftarrow \max(b_{s_i}^a, T_{\max} - a_e^o)$
(7) $b_{s_i}^g \longleftarrow \max(b_{s_i}^g, T_{\max} - a_e^o)$
(8) $b_{s_j}^a \longleftarrow \max(b_{s_j}^a, T_{\max} - a_e^o)$
(9) return $a_s^g, b_s^g, a_s^h, b_s^a$ for $s \in \mathcal{S}$; a_e^o, b_e^o for $e \in \mathcal{E}$.

Algorithm 1: Estimation of truncation points.

Figure 3: Singapore MRT network map.

Table 2: Singapore MRT record samples.

Card id	Type	Boarding time	Alighting time	Origin	Destination
02**5F	Adult	2016-01-25 08:20:04	2016-01-25 08:27:27	35	12
01**1D	Adult	2016-01-25 18:13:57	2016-01-25 18:21:25	12	35
05**3C	Adult	2016-01-26 08:13:51	2016-01-26 08:21:21	35	12
01**8F	Adult	2016-01-26 18:31:45	2016-01-26 18:38:11	12	35

Table 3: Mean square error (MSE) of O-D matrix prediction of the Singapore MRT system.

Prediction horizon	20 min	80 min	120 min
Calendar model	249.11	249.11	249.11
Linear regression with lasso	190.83	280.23	351.28
Linear regression with ridge	191.20	283.76	354.14
ARIMA	2136.98	2315.65	2382.65
LSTM	205.21	228.00	235.10
Random forest	**159.44**	**182.57**	**209.86**

The bold values refer to the best result (or the lowest MSE) of each column.

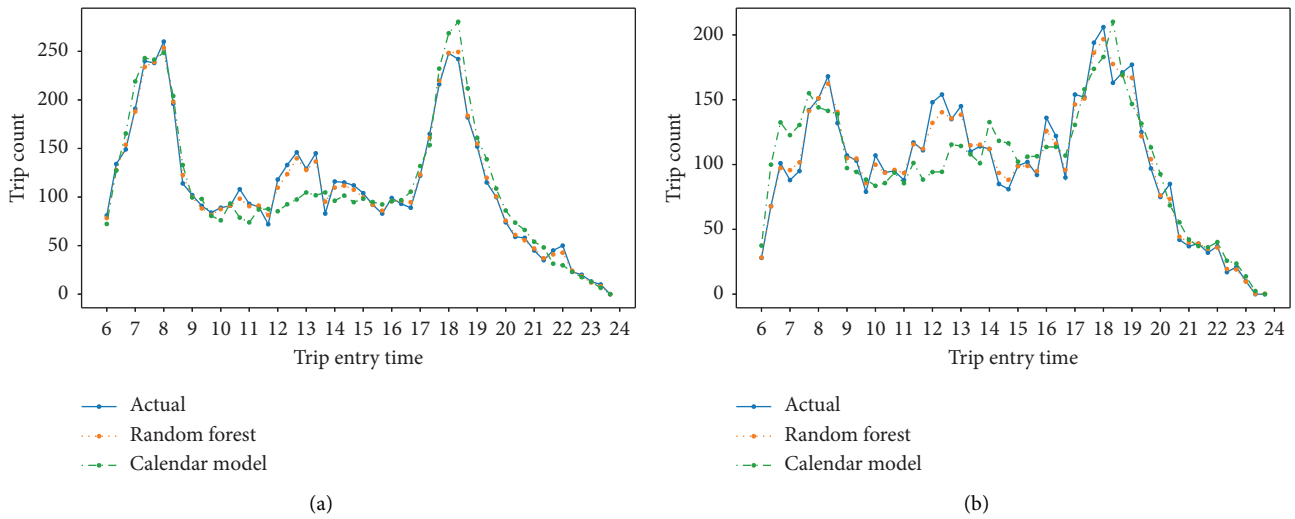


FIGURE 4: One-step-ahead prediction for two selected O-D paths: (a) Boon Lay to Jurong East and (b) Pasir Ris to Tampines.

entry link at Tanah Merah, i.e., $\mu = 7.03$, is relatively larger than that of the other entry links. This is because the Changi Airport branch line of the East-West line has a higher train headway (i.e., 6–9 minutes) than other service lines (i.e., 2–6 minutes). Consequently, passengers at Tanah Merah usually spend more time waiting for trains to Changi Airport. We also notice that transit link (c) takes much longer time than transit link (b). This is because the physical distance between the Expo and Changi Airport is much longer than that between Tanah Merah and Expo.

Based on the estimated travel time distribution of each transit link, we can infer the travel time distribution for each O-D path. The inference results of some O-D paths with single route (i.e., $|M_{ij}| = 1$) are shown in Figures 6(a) and 6(b). The orange dashed line is the probability density function of our estimated truncated Gaussian distribution. The blue line is the empirical probabilistic density function (pdf) of the travel time calculated by kernel density estimation based on the training dataset, which can be regarded as "true" distribution. As observed, the two curves are close to each other, demonstrating that *PIPE* is able to provide a nice fitting of the empirical travel time distribution.

We next present the results corresponding to O-D paths with multiple routes (i.e., $|M_{ij}| > 1$) in Figures 6(c) and 6(d). In this case, we could observe multiple modes in the blue curves. Each mode represents the travel time distribution of one particular route and the magnitude of the mode value is proportional to its route choice probability for a particular entry time (here $\tau = 6$ pm). As observed, the fitted truncated Gaussian mixture models are able to depict both travel time distributions of different routes and their probabilities accurately.

Furthermore, we analyze route preferences of different smart card types, which are captured by the route choice probabilities $\pi_{ij}^{m}(\tau)$ of the truncated Gaussian in the mixture model. Figure 7 shows $\pi_{ij}^{m}(\tau)$ of different smart card types for three selected O-D paths for $\tau = 6$ pm. It shows that for the O-D path from Admiralty to Farrer Park, different smart card types share similar route choice preferences. Most of the passengers prefer route 0, only a small proportion of passengers take route 2, while no passengers are willing to use route 1 to complete their trips. In contrast, for the O-D path from Chinatown to Raffles Place, the route choice preferences vary across smart card types. Both child and student passengers prefer route 0. Yet adults prefer route 1 and old people prefer route 0 and route 1 almost equally. As to the O-D path from Ang Mo Kio to Kent Ridge, almost all the passengers take route 0 and very few passengers take route 1.

We also plot the dynamic route selection for a particular O-D path, i.e., Orchard to Kovan, of different card types. This path has three routes. Route 0 first takes the North-South line from Orchard to Dhoby Ghaut, and then transfers the North-East line to Kovan. Route 1 first takes the North-South line from Orchard to Bishan, then transfers the Circle line to Serangoon, and finally transfers the North-East line to Kovan. Route 2 first takes the North-South line from Orchard to Newton, then transfers the Downtown line to Little India, and lastly transfers the North-East line to Kovan. As shown in Figure 8, the route preferences of different card types vary a lot. For adults who are rushed for time, they take route 0 in the morning and route 2 in the afternoon. This is because the headway of the North-South line is different for its two directions. In the morning, the train headway from Jurong East to Marina Bay is shorter, while since afternoon the train headway from Marina Bay to Jurong East is shorter. Hence route 0 takes a shorter time in the morning, and route 2 takes a shorter time since afternoon. As to route 1, it takes the longest time but is the least crowded, hence old people and children prefer this route.

5.4. Outflow Prediction. Now based on the estimation of the O-D matrix, travel time parameters, and route choice probabilities, we use the remaining 10% dataset to evaluate the prediction performance of *PIPE*. Its prediction MSE for

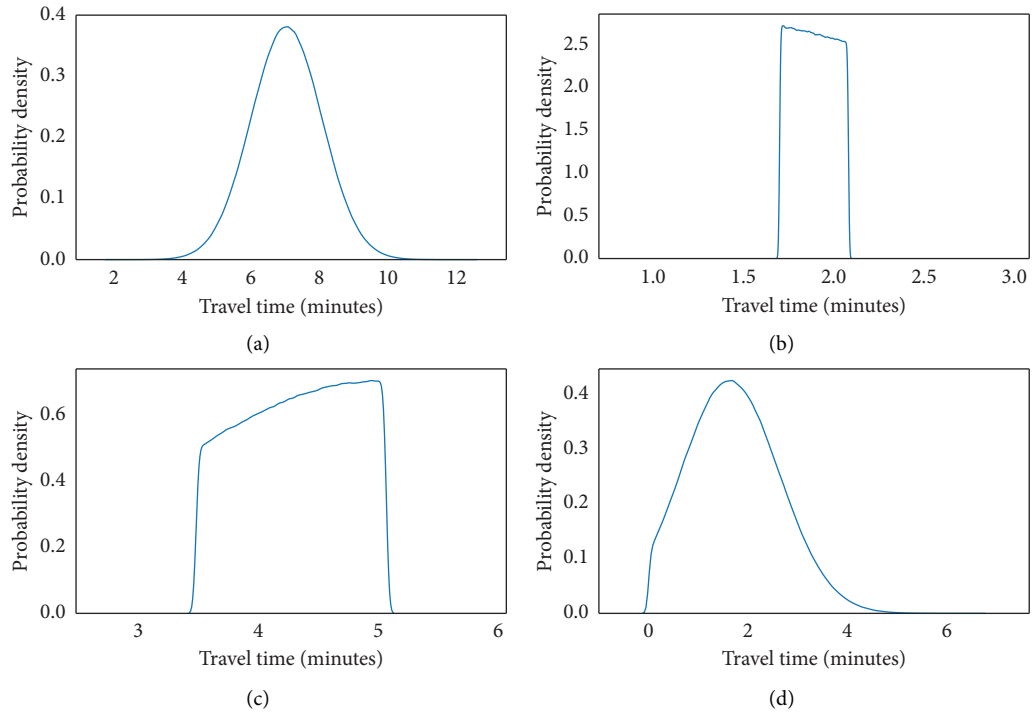


Figure 5: Estimated travel time distribution for transit links of the O-D path from Tanah Merah to Changi Airport.

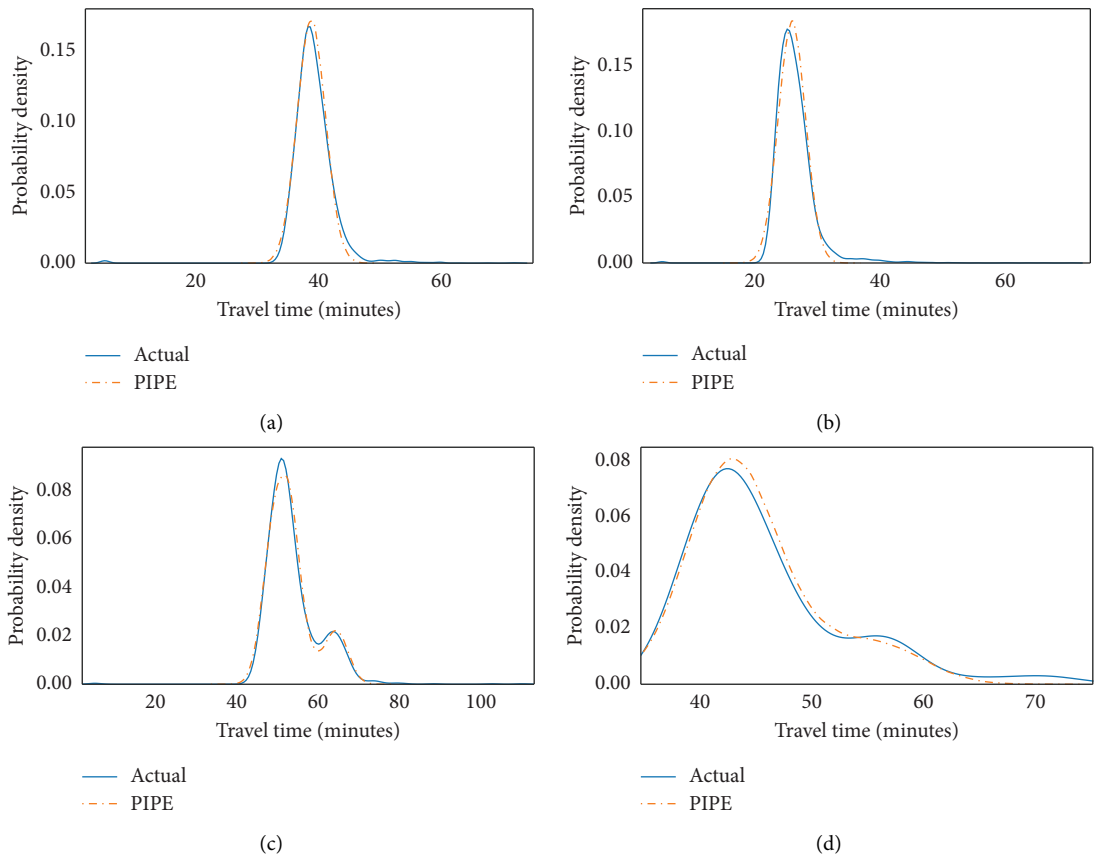


Figure 6: Estimated travel time distribution for a single-route O-D path: (a) Admiralty to Boon Lay and (b) Jurong East to Woodlands. Estimated travel time distribution for a multiroute O-D path: (c) Boon Lay to Toa Payoh and (d) Tampines to Potong Pasir.

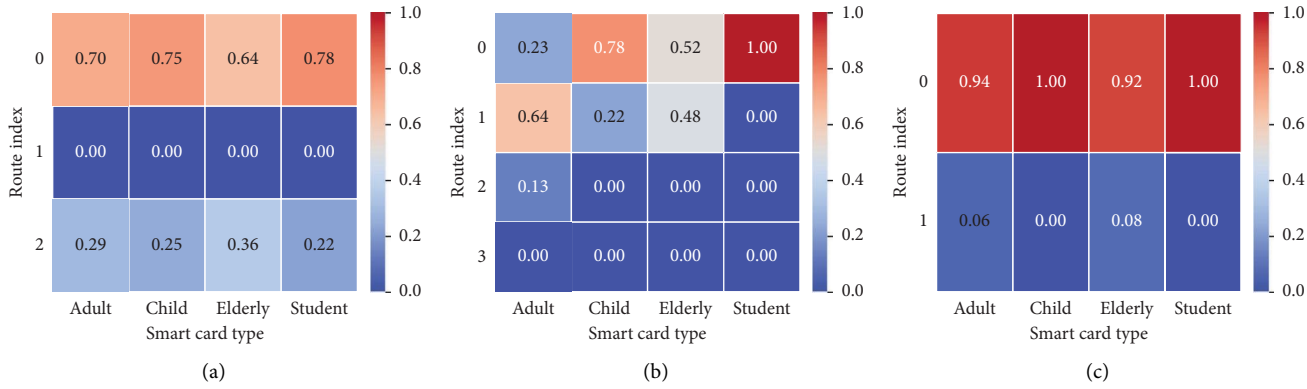


FIGURE 7: Route choice preferences of different smart card types for certain selected O-D paths at τ = 6 pm: (a) Admiralty to Farrer Park, (b) Chinatown to Raffles Place, and (c) Ang Mo Kio to Kent Ridge.

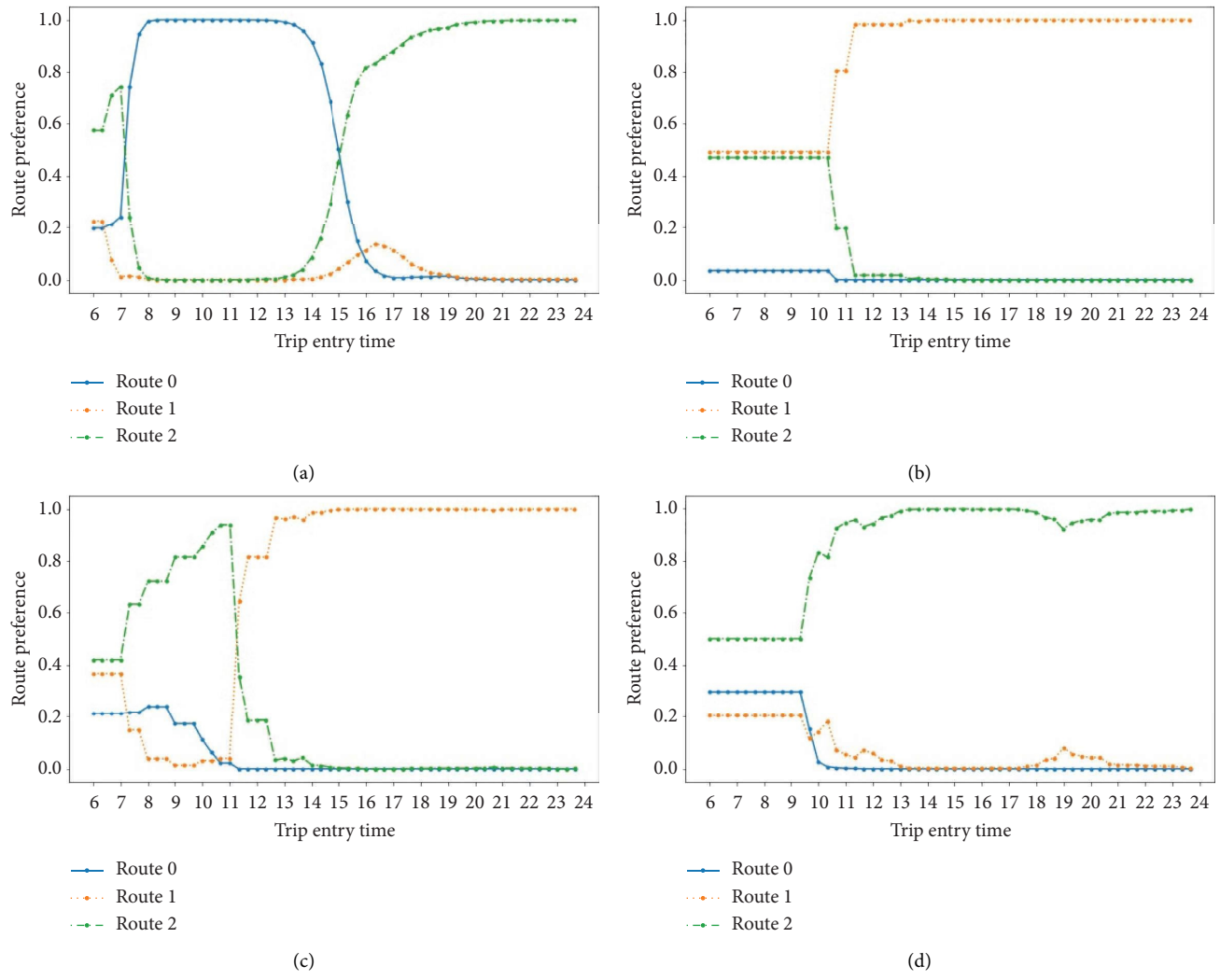


FIGURE 8: Dynamic route choice preferences of different smart card types for the O-D path Orchard to Kovan: (a) adult, (b) child, (c) elderly, and (d) student.

$X_j^{out}(t)$ of each station based on (3) is reported in Table 4. To demonstrate its efficiency, we also compare it with random forest and the calendar model. These two can be regarded as competitive baselines, since they perform the best for the O-D matrix prediction. In particular, for the random forest, it predicts $X_i^{out}(\tau)$ by taking $X_i^{in}(\tau - l), X_i^{out}(\tau - l)$,

Table 4: MSE of station outflow prediction for the Singapore MRT system.

Prediction horizon	20 min	80 min	120 min
Calendar model	6840.64	6840.64	6840.64
Random forest	3328.21	4826.65	5268.05
PIPE	**2433.20**	**2591.15**	**2963.79**

The bold values refer to the best result (or the lowest MSE) of each column.

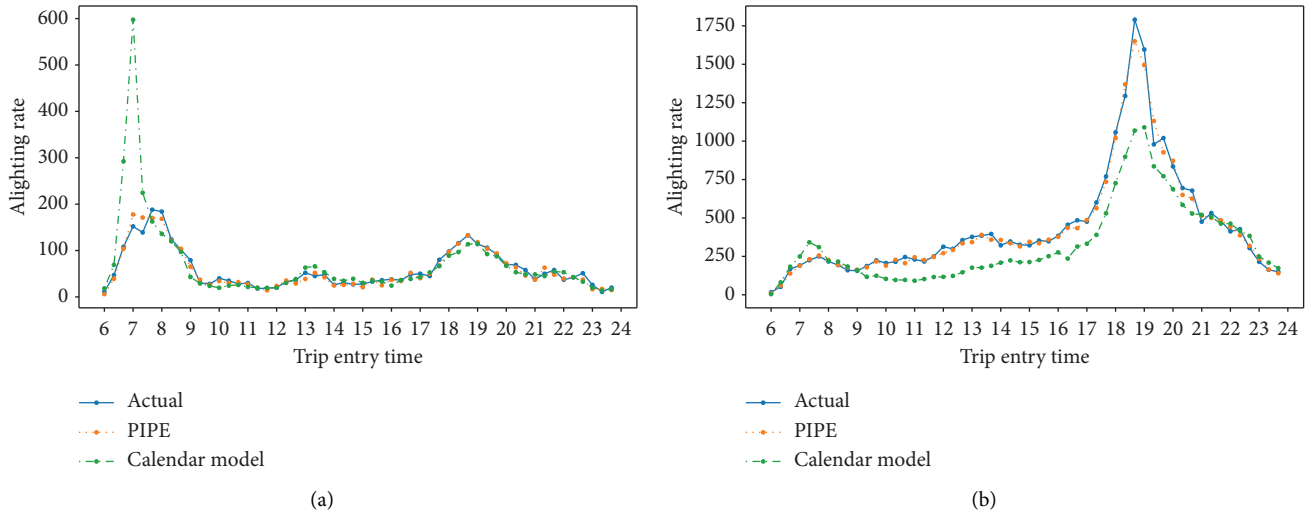
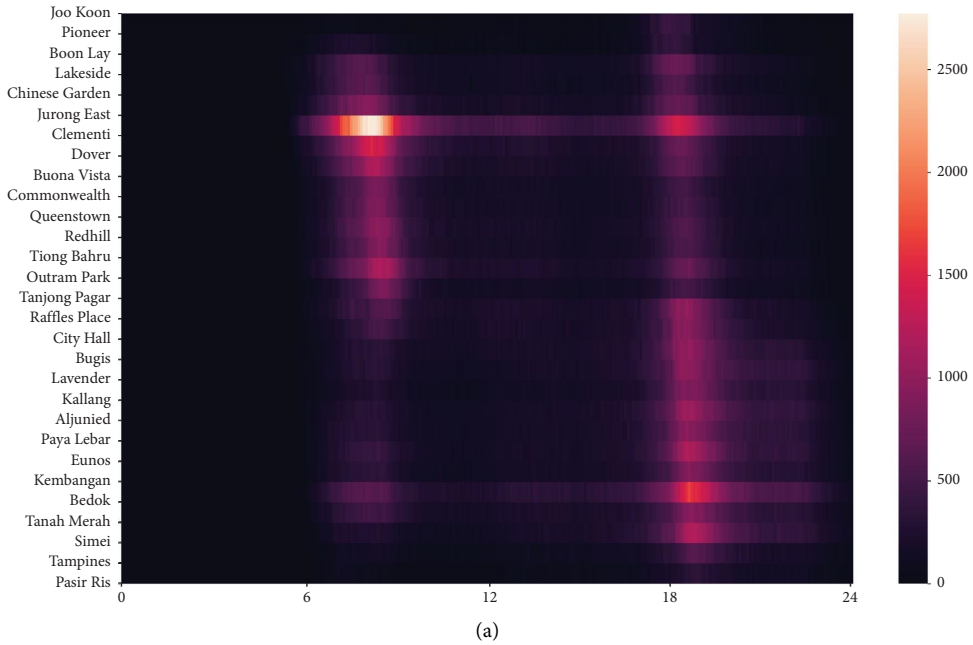
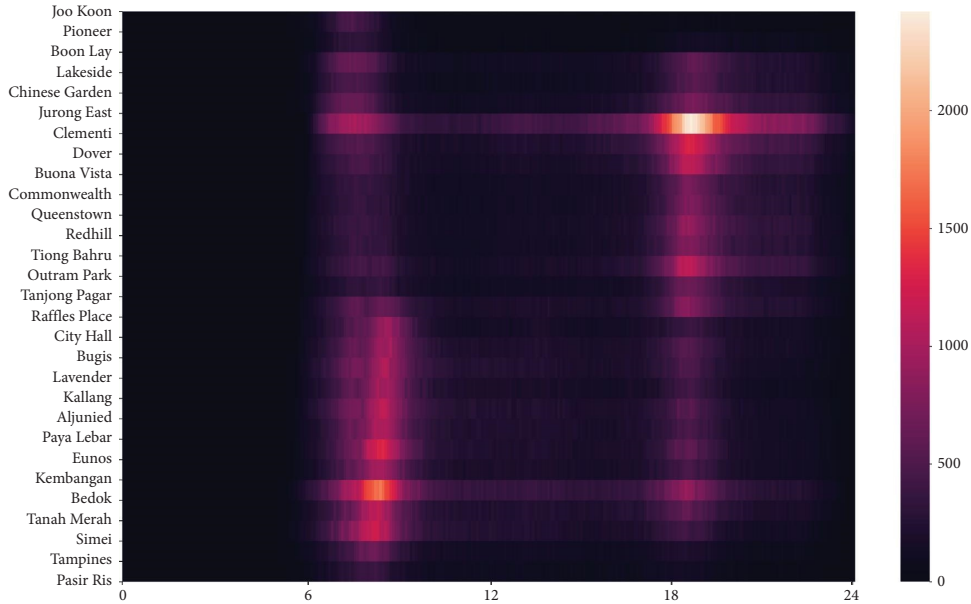


Figure 9: Outflow prediction for (a) Bartley and (b) Punggol.



(a)

Figure 10: Continued.



(b)

FIGURE 10: Passenger load prediction along (a) west ⟶ east direction and (b) east ⟶ west direction of the East-West line over time for the Singapore MRT system.

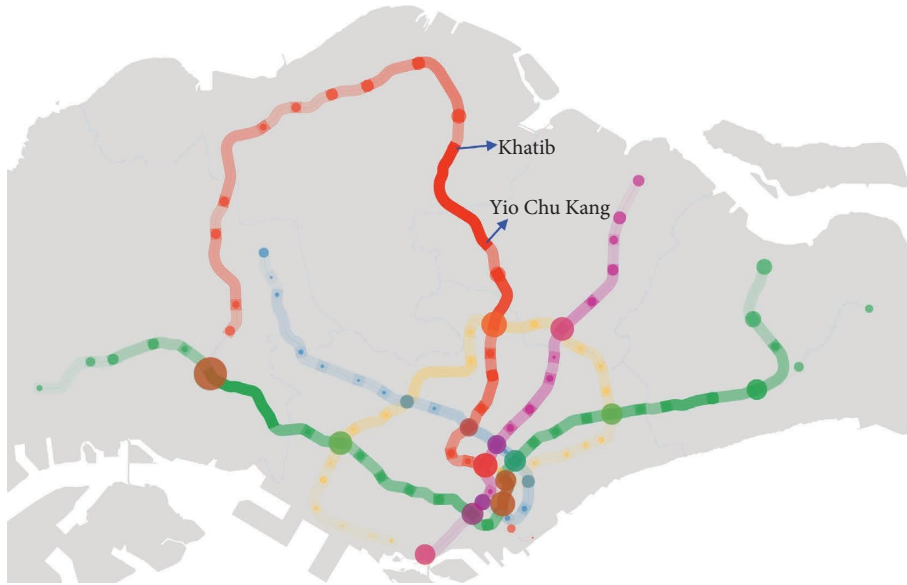


FIGURE 11: Passenger load over the Singapore MRT system at 9 am.

$l = 1, \ldots, \Delta$ as input. This can be regarded as a black box model for outflow prediction, without considering the travel behaviors or traces of passengers inside the metro system. As shown in Table 4, *PIPE* outperforms random forest by a large margin in terms of MSE, which demonstrates the efficacy and advantages of our white-box model. For better illustration, Figure 9 reports the predicted outflow of *PIPE* for two selected stations. The actual outflow together with the prediction result of the calendar model is also reported. As can be observed, *PIPE* can track the small temporal fluctuations of the outflows well, while the calendar model fails to capture lots of details, such as the magnitudes of the peak hour. Though the performance of *PIPE* deteriorates as the prediction horizon increases, *PIPE* still performs consistently better than the calendar method.

5.5. Fine-Grained Passenger Load Prediction. In this section, we present the performance of *PIPE* for passenger load prediction for each metro segment. We choose the most busy line, i.e., the East-West line, to illustrate the passenger load of different metro segments over time. Figure 10 shows the predicted passenger loads along two train directions, i.e., west to east direction and east to west direction. Obviously,

we can observe two morning peaks, one originating from western residential districts (e.g., Jurong East station) all the way to central business districts (CBDs) (e.g., City Hall station) in Figure 10(a), and the other originating from eastern residential areas (e.g., Bedok Station) all the way to CBD in Figure 10(b). These indicate commuting people leave home and make a trip to the office/school at around 8 am.

The other peak hour occurs at about 6 pm, when people get off work and make a trip to home or go to places for entertainment activities such as dinner and shopping. In particular, we can observe two evening peaks, one originating from CBD to eastern residential districts in Figure 10(a), and the other originating from CBD to western residential areas in Figure 10(b). All the abovementioned phenomena actually are quite consistent with common sense and have certain guidelines for metro operators.

We also present the spatial distribution of passenger load for the whole metro system at 9 am for a certain working day in Figure 11. Each node represents a certain station, and the node size indicates the current passenger load inside the MRT station. The color intensity of each metro segment represents its corresponding estimated passenger load. Obviously, there are a few MRT stations much more busy than the others. They are commonly transit hubs where people make transfers or stop by for activities such as dining and entertainment. We also notice that the most crowded segment is the one between Yio Chu Kang and Khatib. This is actually because this segment has the longest distance in the MRT system, and in most cases, there are two trains running on this segment simultaneously. Yet for other segments, most of the time, there is only one single train.

6. Conclusion

In this paper, we propose a statistical inference framework *PIPE* that makes find-grained prediction of passenger load across the metro network. *PIPE* conducts inference tasks including time-dependent O-D matrix forecasting, study of travel time distribution of each transit link inside the metro network, and inference of route choice probabilities. Based on the derived parameters from the inference tasks, *PIPE* is able to predict the passenger load of each metro segment. We apply *PIPE* in the Singapore MRT network, and the satisfactory prediction performance demonstrates its applicability and efficiency.

Data Availability

The data will not be published because of confidential issues with MRT companies.

Disclosure

A preprint has previously been published [34]. This manuscript has been presented as a paper presentation of the *International Journal of Intelligent Systems*.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This paper was supported by the NSFC Grants 71901131, 71932006, and 72271138; the BNSF Grant 9222014; the ASFC Grant 2020Z063058001; and the Tsinghua University Intelligent Logistics Supply Chain Research Center Grant THUCSL20182911756-001.

References

[1] M.-P. Pelletier, M. Trépanier, and C. Morency, "Smart card data use in public transit: a literature review," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 4, pp. 557–568, 2011.

[2] J. Gu, Z. Jiang, J. Chen, and J. Chen, "Short-term trajectory prediction for individual metro passengers integrating diverse mobility patterns with adaptive location-awareness," *Information Sciences*, vol. 599, pp. 25–43, 2022.

[3] G. Vandewiele, P. Colpaert, O. Janssens et al., "Predicting train occupancies based on query logs and external data sources," in *Proceedings of the 26th International Conference on World Wide Web Companion*, Perth, Australia, April 2017.

[4] E. Jenelius, "Data-driven metro train crowding prediction based on real-time load data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2254–2265, 2020.

[5] S. A. Velastin, R. Fernández, J. E. Espinosa, and A. Bay, "Detecting, tracking and counting people getting on/off a metropolitan train using a standard video camera," *Sensors*, vol. 20, no. 21, p. 6251, 2020.

[6] E. Chen, Z. Ye, C. Wang, and M. Xu, "Subway passenger flow prediction for special events using smart card data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1109–1120, 2020.

[7] Y. Gong, Z. Li, J. Zhang, W. Liu, Y. Zheng, and C. Kirsch, "Network-wide crowd flow prediction of sydney trains via customized online non-negative matrix factorization," *International Conference on Knowledge Management*, vol. 23, pp. 1243–1252, 2018.

[8] Y. Sun, B. Leng, and W. Guan, "A novel wavelet-svm short-time passenger flow prediction in beijing subway system," *Neurocomputing*, vol. 166, pp. 109–121, 2015.

[9] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on ITS*, vol. 16, no. 2, pp. 865–873, 2014.

[10] Y. Liu, Z. Liu, and R. Jia, "Deeppf: a deep learning based architecture for metro passenger flow prediction," *Transportation Research Part C: Emerging Technologies*, vol. 101, pp. 18–34, 2019.

[11] S. Hao, D.-H. Lee, and D. Zhao, "Sequence to sequence learning with attention mechanism for short-term passenger flow prediction in large-scale metro system," *Transportation Research Part C: Emerging Technologies*, vol. 107, pp. 287–300, 2019.

[12] Y. Wang, S. Fang, C. Zhang, S. Xiang, and C. Pan, "Tvgcn: time-variant graph convolutional network for traffic forecasting," *Neurocomputing*, vol. 471, pp. 118–129, 2022.

[13] Y. Zhou, J. Li, H. Chen, Y. Wu, J. Wu, and L. Chen, "A spatiotemporal hierarchical attention mechanism-based model for multi-step station-level crowd flow prediction," *Information Sciences*, vol. 544, pp. 308–324, 2021.

[14] X. Ye, S. Fang, F. Sun, C. Zhang, and S. Xiang, "Meta graph transformer: a novel framework for spatial–temporal traffic prediction," *Neurocomputing*, vol. 491, pp. 544–563, 2022.

[15] F. Toqué, E. Côme, M. K. El Mahrsi, and L. Oukhellou, "Forecasting dynamic public transport origin-destination matrices with long-short term memory recurrent neural networks," *Intelligent Transportation Systems Conference*, vol. 52, pp. 1071–1076, 2016.
[16] K.-F. Chu, A. Y. Lam, and V. O. Li, "Deep multi-scale convolutional lstm network for travel demand and origin-destination predictions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 8, pp. 3219–3232, 2020.
[17] H. Shi, Q. Yao, Q. Guo et al., "Predicting origin-destination flow via multi-perspective graph convolutional network," *The International Council for Open and Distance Education*, vol. 56, pp. 1818–1821, 2020.
[18] J. Hu, B. Yang, C. Guo, C. S. Jensen, and H. Xiong, "Stochastic origin-destination matrix forecasting using dual-stage graph convolutional, recurrent neural networks," *The International Council for Open and Distance Education*, vol. 45, pp. 1417–1428, 2020.
[19] H. Peng, H. Wang, B. Du et al., "Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting," *Information Sciences*, vol. 521, pp. 277–290, 2020.
[20] B. He, S. Li, C. Zhang, B. Zheng, and F. Tsung, "Holistic prediction for public transport crowd flows: a spatio dynamic graph network approach," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 321–336, Springer, Berlin, Germany, 2021.
[21] L. Sun, D.-H. Lee, A. Erath, and X. Huang, "Using smart card data to extract passenger's spatio-temporal density and train's trajectory of mrt system," in *SIGKDD International Workshop on Urban Computing*, pp. 142–148, Springer, Berlin, Germany, 2012.
[22] F. Zhang, J. Zhao, C. Tian, C. Xu, X. Liu, and L. Rao, "Spatiotemporal segmentation of metro trips using smart card data," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 3, pp. 1137–1149, 2016.
[23] J. Zhao, F. Zhang, L. Tu et al., "Estimation of passenger route choice pattern using smart card data for complex metro systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 790–801, 2017.
[24] J. Zhang, F. Chen, L. Yang, W. Ma, G. Jin, and Z. Gao, "Network-wide link travel time and station waiting time estimation using automatic fare collection data: a computational graph approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 21034–21049, 2022.
[25] L. Sun, Y. Lu, J. G. Jin, D.-H. Lee, and K. W. Axhausen, "An integrated bayesian approach for passenger flow assignment in metro networks," *Transportation Research Part C: Emerging Technologies*, vol. 52, pp. 116–131, 2015.
[26] X. Xu, L. Xie, H. Li, and L. Qin, "Learning the route choice behavior of subway passengers from afc data," *Expert Systems with Applications*, vol. 95, pp. 324–332, 2018.
[27] X. Tian, B. Zheng, Y. Wang, H.-T. Huang, and C.-C. Hung, "Tripdecoder: study travel time attributes and route preferences of metro systems from smart card data," 2022, https://arxiv.org/abs/2005.01492.
[28] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society-Series B: Statistical Methodology*, vol. 58, no. 1, pp. 267–288, 1996.
[29] A. E. Hoerl and R. W. Kennard, "Ridge regression: biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
[30] A. Liaw and M. Wiener et al.,, "Classification and regression by randomforest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.
[31] S. Makridakis and M. Hibon, "Arma models and the box–jenkins methodology," *Journal of Forecasting*, vol. 16, no. 3, pp. 147–163, 1997.
[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[33] F. Cozman and E. Krotkov, *Truncated Gaussians as Tolerance Sets, Tech. Rep. CMU-RI-TR-94-35*, Carnegie Mellon University, Pittsburgh, PA, USA, 1994.
[34] X. Tian, C. Zhang, and B. Zheng, *Crowding Prediction of In-Situ Metro Passengers Using Smart Card Data*, https://arxiv.org/abs/2009.02880.